

# Top-N recommendations from expressive recommender systems

Cyril J. Stark

Massachusetts Institute of Technology  
77 Massachusetts Avenue, 6-304  
Cambridge, MA 02139-4307, USA  
cyril@mit.edu

November 19, 2015

## Abstract

Normalized nonnegative models assign probability distributions to users and random variables to items; see [Stark, 2015]. Rating an item is regarded as sampling the random variable assigned to the item with respect to the distribution assigned to the user who rates the item. Models of that kind are highly expressive. For instance, using normalized nonnegative models we can understand users' preferences as mixtures of interpretable user stereotypes, and we can arrange properties of users and items in a hierarchical manner. These features would not be useful if the predictive power of normalized nonnegative models was poor. Thus, we analyze here the performance of normalized nonnegative models for top-N recommendation and observe that their performance matches the performance of methods like PureSVD which was introduced in [Cremonesi et al., 2010]. We conclude that normalized nonnegative models not only provide accurate recommendations but they also deliver (for free) representations that are interpretable. We deepen the discussion of normalized nonnegative models by providing further theoretical insights. In particular, we introduce total variational distance as an *operational similarity measure*, we discover scenarios where normalized nonnegative models yield *unique representations* of users and items, we prove that the inference of optimal normalized nonnegative models is *NP-hard* and finally, we discuss the relationship between normalized nonnegative models and nonnegative matrix factorization.

## 1 Introduction

Recommender systems are algorithms that are designed to help users to find interesting items. Hence, good recommender systems are able to predict which items are of interest to which users. At the same time they must be computationally tractable to handle large numbers of users and items. Consequently, recommender systems must have *high predictive power* and they must be *computationally tractable*. These are criteria that necessarily need to be satisfied. However, for some applications we demand more. For instance, the online dating platform OkCupid<sup>1</sup> computes for each user a ‘personality trait’; see figure 1 for an example. These are visual representations that help users to quickly understand the personalities of other users. Thus, these personality traits can be regarded as brief sketches of other users—very much like brief characterizations of movies (e.g., through summary of the plot, list of actors, etc). Hence, visual representations of that kind complement recommendations and help a user to find what they look for.

Expressive visual representations could be drawn directly from the recommender system if the system's representation of users and items is expressive. Therefore we call a recommender system *expressive* if the underlying representations of users and items are highly interpretable. We conclude that for some applications, an ideal recommender system meets the objectives *high predictive power*, *computational tractability* and *high interpretability*.

---

<sup>1</sup><https://www.okcupid.com/>



Figure 1: A part of the author’s personality trait as computed by OkCupid.

How can we address the partially conflicting objectives ‘high predictive power’, ‘computational tractability’ and ‘high interpretability’? In [Stark, 2015] we proposed getting inspiration from engineering and from the natural sciences. There we almost always adopt a paradigm that we might call *system-state-measurement* paradigm. In [Stark, 2015] we adopted that very same perspective for the design of recommender systems. In the context of recommender systems, the *system* is that part of our mind that determines which items we like. The *state* assigned to a user specifies the characteristics of that user’s system, i.e., it describes that user’s taste. The *measurements* we perform on the system to probe the users’ state (i.e., taste) are questions like “Do you like the movie *Ex Machina*?”. Asking many questions of that sort (i.e., performing many measurements) we can get a refined knowledge about a user’s taste (i.e., her state). In large parts of science, the system is modeled by a *sample space*, the state of the system is modeled by a *probability distribution* on that sample space and a measurement is modeled in terms of a random variable whose outcomes are the possible measurement outcomes. These models are called *normalized nonnegative models*; see section 2. Normalized nonnegative models are highly interpretable because these models are identical with the highly interpretable models from the natural sciences and from engineering.

The interpretability of normalized nonnegative models allows to *draw conclusions from user and item representations*. For instance, in [Stark, 2015] we showed how normalized nonnegative models enable us to regard users as mixtures of interpretable user stereotypes and we explained how normalized nonnegative models can be used for the computation of hierarchical orderings of properties of users and items. In sections 4, 5 and 6 we introduce more features of normalized nonnegative models. In particular, we introduce an *operational user-user similarity measure*, we define an *operational item-item similarity measure* and we uncover scenarios where normalized nonnegative models yield *unique representations* of users and items. We complement the discussion of interpretability with an empirical study (cf. section 7) where we test the *predictive power* of normalized nonnegative models. More precisely, we evaluate the performance of these models in terms of *top-N recommendation* where for each user  $u$ , the recommender system must compile a list of  $N$  items that are of interest to that user.

Apart from these practical considerations we investigate the computational complexity of the inference of optimal normalized nonnegative models (section 8) and we explain in what sense normalized nonnegative models are related to nonnegative matrix factorization (section 9).

## Notation

For any  $n \in \mathbb{N}$ ,  $[n] = \{1, \dots, n\}$ .  $\mathbb{R}_+^D$  denotes the set of  $D$ -dimensional nonnegative vectors, i.e.,  $\{\vec{x} \in \mathbb{R}^D | x_j \geq 0\}$ . It contains the probability simplex  $\Delta = \{\vec{x} \in \mathbb{R}_+^D | \sum_j x_j = 1\}$ . By  $\|\vec{v}\|_p$  we denote the  $l_p$  norm of a vector. For any invertible matrix  $A$ ,  $A^{-T} = (A^{-1})^T$ . We will frequently refer to finite sample spaces. These are denoted by  $\Omega = \{\omega_1, \dots, \omega_D\}$  where  $\omega_j$  are elementary events. For any event  $H \subseteq \Omega$  we denote its probability by  $\mathbb{P}[H]$ . By Kolmogorov, a random variable  $\hat{E}$  on  $\Omega$  with alphabet size  $Z$  is a mapping  $\hat{E} : \Omega \rightarrow [Z]$ . We use the notation  $\{\hat{E} = z\} = \hat{E}^{-1}(z) = \{\omega \in \Omega | \hat{E}(\omega) = z\}$ . The total variational distance  $\delta(\vec{p}, \vec{q})$  forms a

natural distance measure between distributions  $\vec{p}, \vec{q} \in \Delta$ . It is defined by  $\delta(\vec{p}, \vec{q}) = \frac{1}{2} \sum_j |p_j - q_j|$ . We use  $U$  to denote the number of users,  $I$  to denote the number of items,  $Z$  the number of different ratings (e.g.,  $Z = 5$  for 5-star ratings).  $R_{ui}$  denotes the rating user  $u$  provides for item  $i$ . The collection of those ratings forms the rating matrix  $R \in [Z]^{U \times I}$ . Recall( $N$ ) is defined in appendix C. We use  $M \subseteq [U] \times [I]$  to mark ratings in the training set. Analogously,  $T$  marks the ratings in the test set.

## 2 Normalized nonnegative models

By Kolmogorov, a (finite) random experiment is defined through the following triple:

- A sample space  $\Omega = \{\omega_1, \dots, \omega_D\}$  where  $\omega_j$  are elementary events.
- A probability measure  $\vec{p} \in \mathbb{R}_+^D$  with  $\sum_j (\vec{p})_j = 1$ .
- A random variable  $\hat{E}$ . This is a function  $\hat{E} : \Omega \rightarrow \{1, \dots, Z\}$  for some  $Z \in \mathbb{N}$ .

The distribution  $\vec{p}$  can be regarded as the *state* of the system under consideration and the random variable  $\hat{E}$  can be regarded as the *measurement* we perform on that system. This interpretation of distributions and random variables is ubiquitous in science and engineering. In [Stark, 2015] we adopt the same perspective for the description of how users rate items. More specifically we denoted by  $R_{ui}$  user  $u$ 's rating of item  $i$ . In case of 5-star-ratings,  $R_{ui} \in \{1, \dots, 5\} = [5]$ , and more generally,  $R_{ui} \in [Z]$  for some  $Z \in \mathbb{N}$ . We described the taste of a user  $u$  in terms of a probability distribution  $\vec{p}_u$  on some (unknown) sample space  $\Omega = \{\omega_1, \dots, \omega_D\}$ , and we regarded the process of asking user  $u$  to rate item  $i$  as a ‘measurement’ we perform on the user’s taste. Therefore, we modeled the question “How do you rate item  $i$ ?” in terms of a random variable  $\hat{E}_i : \Omega \rightarrow [Z]$ . The outcome  $\hat{E}_i$  is the rating of item  $i$ .

Let  $\mathbb{P}_u[\hat{E}_i = z]$  be the probability for user  $u$  to rate item  $i$  with value  $z$ . This probability can be expressed as follows.

$$\mathbb{P}_u[\hat{E}_i = z] = \mathbb{P}_u[\hat{E}_i^{-1}(z)] = \sum_{j=1}^D (\vec{p}_u)_j (\vec{E}_{iz})_j \quad (1)$$

where  $(\vec{E}_{iz})_j = 1$  if  $\omega_j \in \hat{E}_i^{-1}(z)$ , and  $(\vec{E}_{iz})_j = 0$  otherwise (see [Stark, 2015] for examples). By (1), the probabilities  $\mathbb{P}_u[\hat{E}_i = z]$  are determined in terms of an inner product between nonnegative vectors  $\vec{p}_u$  and binary vectors  $\vec{E}_{iz}$  satisfying  $\sum_{z=1}^Z \vec{E}_{iz} = (1, \dots, 1)^T$ . Thus, item  $i$  is described by vectors  $\vec{E}_{i1}, \dots, \vec{E}_{iZ}$  satisfying  $\sum_{z=1}^Z \vec{E}_{iz} = (1, \dots, 1)^T$ . We denote by  $\mathcal{E}$  the set of allowed vectors  $(\vec{E}_{i1}, \dots, \vec{E}_{iZ}) \in \{0, 1\}^{DZ}$ , and we denote by  $\Delta \subset \mathbb{R}_+^D$  the set of all probability distributions  $\vec{p}_u$ . The set  $\Delta$  is convex. However,  $\mathcal{E}$  is not. For computational reasons, we relax  $\mathcal{E}$  to its convex hull  $\mathcal{E}'$ .

We end up with *normalized nonnegative models*: the probability distribution over ratings  $z \in [Z]$  is given by  $(\vec{p}_u^T \vec{E}_{iz})_{z \in [Z]}$  for some vectors  $\vec{p}_u \in \Delta$  and  $(\vec{E}_{i1}, \dots, \vec{E}_{iZ}) \in \mathcal{E}'$ . Here,  $(\vec{E}_{i1}, \dots, \vec{E}_{iZ}) \in \mathcal{E}'$  if and only if

$$\vec{E}_{iz} \in \mathbb{R}_+^D \text{ and } \sum_{z=1}^Z \vec{E}_{iz} = (1, \dots, 1)^T. \quad (2)$$

This class of models is slightly different from conventional probabilistic descriptions because  $\mathcal{E}'$  is equal to the convex hull of the respective set from probability theory. Operationally we can think of the relaxation  $\mathcal{E} \mapsto \mathcal{E}'$  as the result of a two-step procedure. First each user measures her rating for a particular item. Then, the user second-guesses that rating and randomly reassigns the measured rating to another rating. For example a user may actually like an embarrassingly stupid comedy movie. But because she is ashamed of the honest 4-star rating she decides with probability 1/2 to provide a 3-star-rating instead of the honest 4-star rating.

The previous formulation of normalized nonnegative models was *categorical*, i.e., we do not need to assume any scale or linear order of the answers from  $[Z]$ . This feature appears to require a lot of data for

training. Therefore, in section 7, due to the little-data problem, we regard the star ratings as approximate probabilities for liking a particular item, i.e.,

$$\mathbb{P}[u \text{ likes } i] \approx R_{ui}/Z. \quad (3)$$

## 2.1 Algorithm

A natural approach for the computation of normalized nonnegative models (NNM) uses constrained alternating optimization as described in the following algorithm 1. In [Stark, 2015] we comment on the scalability of this approach. Note that all the steps in algorithm 1 can be parallelized and the algorithm converges to a local minimum for root-mean-squared-error on the training set. Note that we fill missing entries in the training set with zeros to counter the selection bias towards popular items. This step is dual to the assumptions the testing procedure (cf. appendix C) relies on, and this step was also employed in [Cremonesi et al., 2010].

---

### Algorithm 1 Constrained least squares

---

- 1: Fix  $D$  (e.g., by cross validation).
  - 2: For all  $u$ , sample  $a_u \in [D]$  uniformly at random and initialize  $\vec{p}_u$  by  $\vec{p}_u = \vec{e}_{a_u}$  where  $(\vec{e}_i)_j = \delta_{ij}$  is a member of the canonical basis.
  - 3: For all  $(u, i) \notin M$ , set  $R_{ui} = 0$  and add  $(u, i)$  to  $M$ .
  - 4: For all items  $i$ , solve the (linearly constrained) nonnegative least squares (NNLS) problem  $\min_{(\vec{E}_{iz})_{z \in [Z]} \in \mathcal{E}'} \sum_{u: (u, i) \in M} (\vec{E}_{iz}^T \vec{p}_u - R_{ui}/Z)^2$ .
  - 5: For all users  $u$ , solve the (linearly constrained) nonnegative least squares (NNLS) problem  $\min_{\vec{p}_u \in \Delta} \sum_{i: (u, i) \in M} (\vec{E}_{iz}^T \vec{p}_u - R_{ui}/Z)^2$ .
  - 6: Repeat steps 4 and 5 until a stopping criteria is satisfied (e.g., maximum number of iterations).
- 

## 3 Limited interpretability of general matrix factorizations

In the most basic matrix factorization models (see for example [Koren et al., 2009]) we assign vectors  $\vec{x}_u \in \mathbb{R}^D$  to users  $u$  and we assign vectors  $\vec{y}_i \in \mathbb{R}^D$  to items  $i$ . These vectors are chosen such that  $\vec{x}_u^T \vec{y}_i \approx R_{ui}$ . To understand limitations of that approach let us assume for simplicity that the matrix factorization technique we employ is unregularized. I.e., every family of vectors  $\vec{x}_u, \vec{y}_i \in \mathbb{R}^D$  provides valid descriptions of the users and items as long as  $\vec{x}_u^T \vec{y}_i \approx R_{ui}$ . Apart from overfitting there is another problem with unregularized factorizations of that type: for any invertible matrix  $A$  we have

$$\vec{x}_u^T \vec{y}_i \approx R_{ui} \Leftrightarrow (A^{-T} \vec{x}_u)^T A \vec{y}_i \approx R_{ui}. \quad (4)$$

Hence, there are many equivalent ways to represent users and items in terms of vectors  $\vec{x}_u$  and  $\vec{y}_i$ , respectively. This freedom significantly *limits the possibility to base any interpretation of the users' behavior on geometric properties of  $\vec{x}_u$  and  $\vec{y}_i$ .*

For instance, we cannot use  $\|\vec{x}_u - \vec{x}_{u'}\|$  as user-user similarity (important in collaborative filtering [Resnick et al., 1994, Sarwar et al., 2001, Deshpande and Karypis, 2004, O Connor and Herlocker, 1999, Sarwar et al., 2002]. To illustrate this point, we choose  $A$  proportional to the identity matrix  $I$ , i.e.,  $A = \lambda I$  for some  $\lambda \neq 0$ . It follows that the variation of  $\lambda$  allows to choose  $\|A^{-T} \vec{x}_u - A^{-T} \vec{x}_{u'}\|$  arbitrarily because

$$\|A^{-T} \vec{x}_u - A^{-T} \vec{x}_{u'}\| = \|\vec{x}_u - \vec{x}_{u'}\|/\lambda. \quad (5)$$

Observation (5) is independent of the norm  $\|\cdot\|$  we choose because  $\|\vec{v}/\lambda\| = \|\vec{v}\|/\lambda$  for all norms. Another popular user-user similarity measure is cosine similarity [Resnick et al., 1994, Herlocker et al., 2002, McLaughlin and Herlocker, 2004, Sarwar et al., 2001]. To explain why this similarity measure is not stable under the transformations (4), we assume  $\vec{x}_u = (1, 0)$  and  $\vec{x}_{u'} = (1, \varepsilon)$  for some small scalar  $\varepsilon > 0$ . It

follows that  $\angle(\vec{x}_u, \vec{x}_{u'}) \approx 0$  where  $\angle(\vec{x}_u, \vec{x}_{u'})$  denotes the angle between  $\vec{x}_u$  and  $\vec{x}_{u'}$ . However, choosing  $A = \text{diag}(1, \lambda)$  with  $\lambda \gg 1/\epsilon$ , we get  $\angle(A^{-T}\vec{x}_u, A^{-T}\vec{x}_{u'}) \approx \pi/2$ . Choosing  $A$  more generally, we can squeeze and stretch the vectors  $\{\vec{x}_u\}_{u \in [U]}$  in arbitrary directions.

Obviously, regularizing the matrix factorization by penalizing the norms of the vectors  $\vec{x}_u$  and  $\vec{y}_i$  (e.g., Tikhonov regularization) helps to restrict our freedom (4) to rescale user and item vectors. However, even if the solution of problems of the sort

$$\min \left( \sum_{u,i} (\vec{x}_u^T \vec{y}_i - R_{ui})^2 \right) + \mu \left( \sum_u \|\vec{y}_u\|^2 \right) + \mu \left( \sum_i \|\vec{y}_i\|^2 \right)$$

is unique we still do not know how to interpret the vectors  $\vec{x}_u$  and  $\vec{y}_i$ . We do not even know whether  $\|\vec{x}_u - \vec{x}_{u'}\|$  accurately reflects the actual similarity of users  $u$  and  $u'$ . In fact it is easy to come up with two regularization procedures that both guarantee uniqueness but lead to disagreeing distances between vectors we assign to users. To summarize we note that

- different regularization schemes may or may not lead to unique determination of geometric quantities like  $\|\vec{x}_u - \vec{x}_{u'}\|$  that we wish to interpret operationally.
- But even among the regularization schemes that uniquely determine geometric quantities like  $\|\vec{x}_u - \vec{x}_{u'}\|$ , a change of the regularization leads to a change of the values  $\|\vec{x}_u - \vec{x}_{u'}\|$ .

Hence, regularization, geometric interpretability and similarity are *intimately related to each other*. Non-negative matrix factorization (NMF) has become popular because of the interpretability of the user and item vectors. However, NMF without norm penalization (Tikhonov) suffers from the same issues mentioned before as the transformation  $A$  from our examples maps vectors from  $\mathbb{R}_+^D$  back into  $\mathbb{R}_+^D$ . NMF is particularly tricky as the freedom described by  $A$  cannot only be used to alter similarity measures, it can also be used to change our interpretation of the user and item vectors. For instance, a positive matrix  $A = \text{diag}(\lambda_1, \dots, \lambda_D)$  can be used to arbitrarily shrink or stretch vectors  $\vec{x}_u$  in all directions. Consequently, in one NMF we might have  $\vec{x}_u = (1, \epsilon, \dots, \epsilon)^T$  leading to the conclusion that user  $u$  is very well described by the feature associated with  $(1, 0, \dots, 0)^T$ . On the other hand, for  $A = \text{diag}(1/\epsilon, 1, \dots, 1, \epsilon)$  we have that  $A^{-T}\vec{x}_u = (\epsilon, \dots, \epsilon, 1)^T$  leading to the conclusion that user  $u$  is accurately described by the feature represented by  $(0, \dots, 0, 1)^T$ .

## 4 Uniqueness

We claim that ambiguities of the form (4) are addressed in normalized nonnegative models. Intuitively, one expects the user vectors  $\vec{p}_u$  and the item vectors  $\vec{E}_{iz}$  to be approximately uniquely defined if the entries in the rating matrix  $R$  push these vectors towards the boundary of the cone  $\mathbb{R}_+^D$ . That is because in these situations, the set of allowed states  $\Delta$  and the set of allowed measurements  $\mathcal{E}'$  leave no room to wiggle the user and item vectors as in (4). To confirm this intuition we consider an toy example that we can analyze rigorously. Assume that  $D = Z$  and assume that the users do not just provide  $R$ . Instead, for each  $u, i, z$ , user  $u$  provides us with an accurate estimate of the probability to rate  $i$  with  $z$  ‘stars’. Moreover, we assume that for each  $i, z$  there exists at least one user  $u_{iz}$  who rates  $i$  with  $z$  ‘stars’. We claim that datasets of that kind uniquely determine the underlying normalized nonnegative model.

To prove this claim we first observe that by Cauchy-Schwarz,

$$1 = \mathbb{P}_{u_{iz}}[\hat{E}_i = z] = \vec{p}_{u_{iz}}^T \vec{E}_{iz} \leq \|\vec{p}_u\|_2 \|\vec{E}_{iz}\|_2 \leq \|\vec{E}_{iz}\|_2 \quad (6)$$

for all  $i, z$  because  $\|\vec{p}\|_2 \leq 1$  for every probability distribution. For any  $\vec{E}_{iz}, \vec{E}_{iz'} \in \mathbb{R}_+^D$  we have that  $\vec{E}_{iz}^T \vec{E}_{iz'} \geq 0$ . Therefore, using  $\|\vec{v}\|_2^2 = \vec{v}^T \vec{v}$ ,

$$D = \|(1, \dots, 1)^T\|_2^2 = \sum_z \|\vec{E}_{iz}\|_2^2 \geq \sum_z \vec{E}_{iz}^T \vec{E}_{iz} \geq Z = D. \quad (7)$$

where we used (6) in the last inequality. Equation (7) can only be satisfied if  $\vec{E}_{iz}^T \vec{E}_{iz'} = \delta_{zz'}$ . This in turn can only be satisfied if each of the *nonnegative* vectors  $\vec{E}_{iz}$  is equal to an element of the orthonormal basis defining  $\mathbb{R}_+^D$ . It follows that  $\vec{p}_{u_{iz}} = \vec{E}_{iz}$  because this is the only possibility to satisfy  $1 = \mathbb{P}_{u_{iz}}[\hat{E}_i = z]$ . This concludes the proof of the claim.

We note that this sequence of arguments crucially depends on both the *conic structure* (i.e.,  $\mathbb{R}_+^D$ ) of NNMs, as well as on the *normalization conditions* of normalized nonnegative models.

## 5 Operational user similarity

Assume we describe users and items in terms of a normalized nonnegative model. In this section we are going to motivate the use of the total variational distance  $\delta(\vec{p}_x, \vec{p}_{x'}) = \frac{1}{2} \sum_{j=1}^D |(\vec{p}_x)_j - (\vec{p}_{x'})_j|$  as user-user similarity measure by a game which provides an operational interpretation of  $\delta(\vec{p}_x, \vec{p}_{x'})$ . Imagine two users  $u = 1, u = 2$  and an item  $i$ . Assume you know the representations  $\vec{p}_1, \vec{p}_2 \in \Delta$  of the users' tastes and assume you know the description  $(\vec{E}_{i1}, \dots, \vec{E}_{iZ}) \in \mathcal{E}$  of item  $i$ . Now

1. a referee flips an unbiased coin to select a user  $\hat{u} \in \{1, 2\}$ .
2. The referee asks user  $\hat{u}$  to rate item  $i$ . We denote the value of  $\hat{u}$ 's rating by  $r$ .
3. Then, the referee hands you a note specifying  $r$  but not  $\hat{u}$ .
4. Your objective is to guess  $\hat{u}$ .

We can compute all the probabilities  $\mathbb{P}_u[\hat{E}_i = r]$  for user  $u \in \{1, 2\}$  to rate  $i$  with  $r$  because we know the descriptions  $\vec{p}_1, \vec{p}_2$  of the users and the description  $(\vec{E}_{i1}, \dots, \vec{E}_{iZ})$  of the item. If  $\mathbb{P}_1[\hat{E}_i = r] > \mathbb{P}_2[\hat{E}_i = r]$  then we better guess that user 1 provided the rating. Otherwise, we guess that user 2 provided the rating. Set  $\mathcal{Z} = \{z \in [Z] \mid \mathbb{P}_1[\hat{E}_i = z] > \mathbb{P}_2[\hat{E}_i = z]\}$ . Hence, we guess  $\hat{u} = 1$  if and only if  $r \in \mathcal{Z}$ . This (optimal) strategy succeeds with probability  $p_{\text{success}} = \frac{1}{2} \mathbb{P}_1[\hat{E}_i \in \mathcal{Z}] + \frac{1}{2} \mathbb{P}_2[\hat{E}_i \notin \mathcal{Z}]$  (recall that the coin is unbiased). By complementarity of the events  $\{\hat{E}_i \in \mathcal{Z}\}$  and  $\{\hat{E}_i \notin \mathcal{Z}\}$ ,

$$p_{\text{success}} = \frac{1}{2} \left( 1 + \mathbb{P}_1[\hat{E}_i \in \mathcal{Z}] - \mathbb{P}_2[\hat{E}_i \in \mathcal{Z}] \right) \leq \frac{1}{2} \left( 1 + \delta(\vec{p}_1, \vec{p}_2) \right) \quad (8)$$

where  $\delta(\vec{p}_1, \vec{p}_2)$  denotes the *total variational distance*, i.e.,

$$\delta(\vec{p}_1, \vec{p}_2) = \max_{\Omega_0 \subseteq \Omega} |\mathbb{P}_1[\Omega_0] - \mathbb{P}_2[\Omega_0]| = \frac{1}{2} \|\vec{p}_1 - \vec{p}_2\|_1 \quad (9)$$

(see [Cover and Thomas, 2012]). We conclude that  $\delta(\vec{p}_1, \vec{p}_2)$  yields an upper bound on our success probability which is independent of the item  $i$ . More importantly, however, this upper bound is tight, meaning that there exists a hypothetical item  $i^*$  that leads to a success probability  $p_{\text{success}}^*$  satisfying

$$p_{\text{success}}^* = \frac{1}{2} \left( 1 + \delta(\vec{p}_1, \vec{p}_2) \right). \quad (10)$$

The item  $i^*$  is any item satisfying  $\{\hat{E}_i \in \mathcal{Z}\} = \Omega^*$  where  $\Omega^*$  is the maximizer from (9). Identity (10) captures the operational meaning of the total variational distance:  $\delta(\vec{p}_1, \vec{p}_2)$  determines via (10) the maximal success probability for distinguishing the users  $u_1$  and  $u_2$ . *This motivates using  $1 - \delta(\vec{p}_1, \vec{p}_2)$  as similarity measure because  $\delta(\vec{p}_1, \vec{p}_2)$  is small if and only if users  $u_1, u_2$  are difficult to distinguish.* In appendix B we remind the reader of an alternative interpretation of  $\delta(\vec{p}_1, \vec{p}_2)$ . We expect  $\delta(\vec{p}_1, \vec{p}_2)$  to be particularly useful to create matches on dating websites.

## 6 Operational item similarity

To derive an operational item-item similarity measure we consider a game similar to the game from section 5. More precisely, assume you know the representations  $(\vec{E}_{11}, \dots, \vec{E}_{1Z}), (\vec{E}_{21}, \dots, \vec{E}_{2Z})$  of two items  $i = 1, 2$  and assume you know the representation  $\vec{p}$  of a user  $u$ . The game proceeds as follows.

1. A referee flips an unbiased coin to select an item  $\hat{i} \in \{1, 2\}$ .
2. The referee secretly asks user  $u$  to rate item  $\hat{i}$ . We denote the value of that rating by  $r$ .
3. The referee hands you a note specifying  $r$  but not  $\hat{i}$ .
4. Your objective is to guess  $\hat{i} \in \{1, 2\}$ .

Knowing  $\vec{p}$  and both item representations, we can compute the two distributions  $\vec{q} := (\mathbb{P}[r|\hat{i} = 1])_{r \in [Z]}$  and  $\vec{q}' := (\mathbb{P}[r|\hat{i} = 2])_{r \in [Z]}$ . Hence, we can rephrase the considered game: we get the value  $r$  sampled from either  $\vec{q}$  or  $\vec{q}'$ . Our task is to guess which of the two alternatives applies. According to section 5,  $p_{\text{success}} = \frac{1}{2}(1 + \frac{1}{2}\|\vec{q} - \vec{q}'\|_1)$  (‘=’ because this time we can choose the strategy freely). What is the optimal success probability when varying  $\vec{p}$ ? Note that

$$\begin{aligned} & \max_{\vec{p} \in \Delta} \|\vec{q} - \vec{q}'\|_1 \text{ s.t. } \left\{ q_z = \vec{p}^T \vec{E}_{1z}, q'_z = \vec{p}^T \vec{E}_{2z} \forall z \in [Z] \right\} \\ &= \max_{j \in [D]} \|\vec{q} - \vec{q}'\|_1 \text{ s.t. } \left\{ q_z = (\vec{E}_{1z})_j, q'_z = (\vec{E}_{2z})_j \forall z \in [Z] \right\}. \end{aligned}$$

Here, we used that maximization of  $\|\cdot\|_1$  is a LP and therefore achieved at an extremal point of  $\Delta$ . Therefore, we end up with the item-item similarity measure  $1 - \delta(i_1, i_2)$  where

$$\delta(i_1, i_2) = \frac{1}{2} \max_{j \in [D]} \|(E_1 - E_2)_{j,:}\|_1 \quad (11)$$

with  $E_i := (\vec{E}_{i1}, \dots, \vec{E}_{iZ})$ ;  $i = 1, 2$  and  $(E_1 - E_2)_{j,:}$  denotes the  $j$ -th row of  $E_1 - E_2$ . Using (11), we show in appendix A how *all* applications from [Stark, 2015] can be lifted to situations where the available data does not specify any item tags.

## 7 Empirical study

Running Algorithm 1, we evaluate the performance of normalized nonnegative models on the MovieLens 1M dataset from [Miller et al., 2003]. This allows us to compare our results with the results obtained in the literature; e.g., [Cremonesi et al., 2010]. We are interested in the part of the MovieLens dataset which specifies a long list of triples  $(u, i, R_{ui})$  where  $u$  is a user,  $i$  is an item (i.e., a movie) and  $R_{ui} \in [5]$  is the 5-star-rating of  $i$  by  $u$ . To define the training data and the test data we proceed exactly as in [Cremonesi et al., 2010], i.e., we randomly sample 1.4% of the provided movie ratings. These ratings form the test set  $T$ . The remaining entries form the training set  $M$ . In appendices C and D we remind the reader of the definition of recall at  $N$ , and we sketch the evaluation methodology [Cremonesi et al., 2010] which we adopt in the following.

Figure 2 (left) compares normalized nonnegative models (computed using 10 iterations) with some common recommender systems (see [Cremonesi et al., 2010] for details). We observe that normalized nonnegative models perform particularly well for low  $N$ . This might be of interest in applications because we do not want to present long lists of recommendations to users. Figure 2 (right) also compares normalized nonnegative models with other recommender systems. This time, however, we only take into account items from the long tail for the evaluation of recall (cf. appendix D). Finally, figure 3 displays recall at 20 as function of both the dimension of the model or as function of the iteration (in Algorithm 1). All of these results were computed on a desktop computer (4 cores) running Matlab.

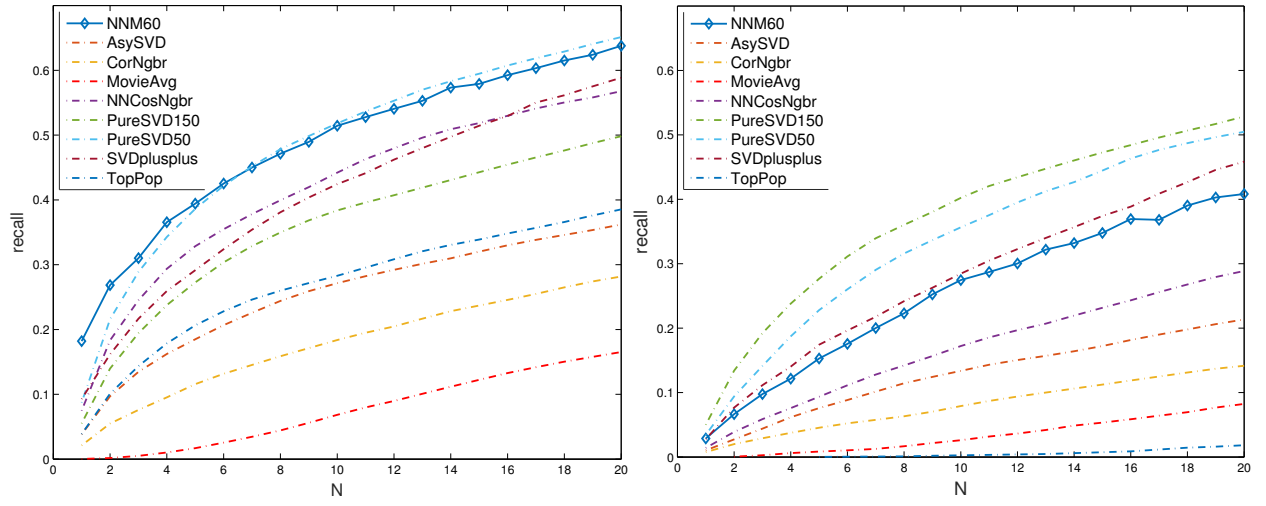


Figure 2: *Left*: Recall at  $N$ ; all items. *Right*: Recall at  $N$ ; items from long tail.

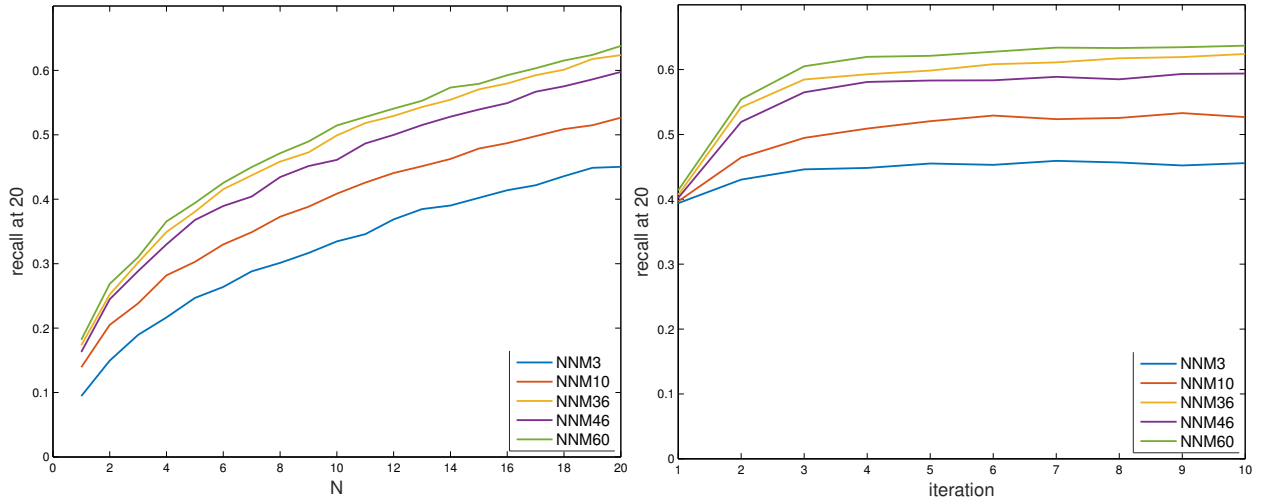


Figure 3: *Left*: Recall at  $N$  (all items); comparison of different model dimensions. *Right*: Recall at 20 (all items) as function of iteration.



## 8 Computational complexity

We consider the categorical setting described in equation (1) and imagine that (instead of star ratings) the users provide estimates for  $\mathbb{P}_u[\hat{E}_i = z]$ . What is the computational complexity of finding the lowest-dimensional normalized nonnegative model for the data  $\mathbb{P}_u[\hat{E}_i = z]$ ? In other words, what is the computational complexity of problem *MinDim* defined as follows.

*MinDim.* Find the minimal dimension  $D$  such that there exist  $\vec{p}_u \in \Delta$  and  $\vec{E}_{iz} \in \mathcal{E}'$  with the property  $\vec{p}_u^T \vec{E}_{iz} = \mathbb{P}_u[\hat{E}_i = z]$  for all  $u, i, z$ .

In appendix E we prove the following theorem 3 by showing that the natural decision version of *MinDim* is NP-hard.

**Theorem 1.** *The decision problem MinDim is NP-hard.*

## 9 Relation to nonnegative matrix factorization

Assume we are dropping all the normalization constraints on  $\vec{p}_u$  and  $\vec{E}_{iz}$ . I.e., instead of searching for a NNM with the property  $\mathbb{P}_u[\hat{E}_i = z] = \vec{p}_u^T \vec{E}_{iz}$  we search for *arbitrary* nonnegative vectors  $\vec{a}_u, \vec{b}_{iz} \in \mathbb{R}_+^D$  satisfying  $\mathbb{P}_u[\hat{E}_i = z] = \vec{a}_u^T \vec{b}_{iz}$ . The following Lemma 2 characterizes the relationship between models  $\vec{a}_u, \vec{b}_{iz}$  for  $\mathbb{P}_u[\hat{E}_i = z]$  on the one hand and NNMs for  $\mathbb{P}_u[\hat{E}_i = z]$  on the other hand.

**Lemma 2.** *Let  $A \in \mathbb{R}_+^{D \times U}$ ,  $B \in \mathbb{R}_+^{D \times IZ}$  be arbitrary nonnegative matrices with the properties*

$$(\mathbb{P}_u[\hat{E}_i = z])_{u \in [U]; i \in [I], z \in [Z]} = A^T B \quad (12)$$

*and  $D = \text{rank}(A) = \text{rank}(B)$ . We denote by  $\vec{a}_u$  the columns of  $A$  and by  $\vec{b}_{iz}$  the columns of  $B$ . Then, there exists an invertible matrix  $T$  such that*

$$\vec{a}'_u := T \vec{a}_u, \quad \vec{b}'_{iz} := T^{-1} \vec{b}_{iz}$$

*is a normalized nonnegative model satisfying  $\mathbb{P}_u[\hat{E}_i = z] = \vec{a}'_u{}^T \vec{b}'_{iz}$  for all  $u, i, z$ .*

Factorizations (12) are of great importance in many disciplines; factorizations of that type are called nonnegative matrix factorizations (NMF; [Lee and Seung, 1999]). Note, however, that Lemma 2 does not imply a general equivalence between NMF and NNM because to establish the transformation from NMF to NNMs and vice versa we considered the noiseless scenario and we assumed that  $D = \text{rank}(A) = \text{rank}(B)$  (see Lemma 2). We prove Lemma 2 in appendix F.

## 10 Related work

We mentioned related work when discussing similarity measures, time complexity and relation to NMF. Therefore, we describe here relations between NNMs and other models for item recommendation. We start with *Probabilistic matrix factorization* (PMF, see [Mnih and Salakhutdinov, 2007, Salakhutdinov and Mnih, 2008]) which forms intriguing family of models related to NNMs. In PMFs, the rating  $R_{ui}$  of user  $u$  for item  $i$  is regarded as Gaussian random variable. Mean and variance of  $R_{ui}$  are modeled in terms of  $\vec{U}_u^T \vec{V}_i$  and  $\sigma$ , respectively. Here,  $\vec{U}_u, \vec{V}_i$  are low-dimensional vectors assigned to users and items. This structure is reminiscent of (1) and (13). However, to apply PMF we need to assume that  $R_{ui}$  is Gaussian. In contrast, to apply NNMs, we do not need to assume anything about the distribution of  $R_{ui}$  and  $R_{ui}$  can be treated as *categorical* random variable. The interpretability of PMFs is high because in principle, they allow for the computation of hierarchical orderings of properties of users and items (through the game introduced in [Stark, 2015]). But due to the infinite-dimensional nature of PMFs (i.e.,  $|\Omega| = \infty$ ), the behavior of users cannot be interpreted easily through stereotypes; see [Stark, 2015].

In applications like top- $N$  recommendation [Herlocker et al., 2004] we are not primarily interested in ratings of items but we are interested only in the co-occurrence of pairs  $(u, i)$  in measured data. The occurrence of a pair  $(u, i)$  can be interpreted, for instance, as “ $u$  likes movie  $i$ ”, “ $u$  attends concert  $i$ ”, etc. When using the graphical *aspect models* [Hofmann and Puzicha, 1999, Hofmann, 1999, Blei et al., 2003, Blei et al., 2004] to describe these settings we regard  $(u, i)$  as a two-dimensional random variable whose distribution has the form

$$\mathbb{P}[u, i] = \sum_{k=1}^K \mathbb{P}[u|k] \mathbb{P}[i|k]. \quad (13)$$

Hence, aspect models are described in terms of a latent variable  $k \in [K]$  and we observe that  $u$  and  $i$  are independent when conditioned on  $k$ . By (13),  $\mathbb{P}[u, i]$  can be regarded as inner product between two distributions (i.e.,  $(\mathbb{P}[u|k])_{k \in [K]}$  and  $(\mathbb{P}[i|k])_{k \in [K]}$ ) on some sample space  $\Omega = \{\omega_1, \dots, \omega_K\}$ . In that sense, aspect models (13) are related to normalized nonnegative models (1). The difference lies in the *different interpretation* of the vectors whose inner product we compute. In case of normalized nonnegative models we compute the inner product between a distribution (describing the user) and a convexly relaxed indicator function (describing one particular rating  $z$  of the item). In case of aspect models we compute the inner product between two distributions—the first distribution describes the user and the second distribution describes the item. In practice, this distinction reveals itself (for example) when we try to use aspect models to model 5-star ratings (an instance of ratings with multiple outcomes). Describing such ratings with normalized nonnegative models is straightforward (cf. section 2). On the other hand, describing ratings with multiple outcomes with aspect models is more involved because we need to decide on a particular graphical model (cf. section 2.3 in [Hofmann and Puzicha, 1999]) and this makes the practical application of aspect models more challenging. Moreover, to the best of our knowledge, it has not yet been discussed carefully in what precise sense aspect models give rise to operational similarity measures like the total variational distance.

Regarding the empirical evaluation of top- $N$  recommendation we would like to point out the closely related works [Cremonesi et al., 2010] and [Barbieri and Manco, 2011]. Both of these beautiful works show the disagreement between recall and precision on the one hand and RMSE on the other hand. The paper [Cremonesi et al., 2010] introduces PureSVD for top- $N$  recommendation. Moreover, it proposes a construction of the test set that we adopt here, namely, the exclusion of most popular items to counteract the selection bias in the MovieLens dataset. The paper [Barbieri and Manco, 2011] analyzes the performance of major probabilistic models for top- $N$  recommendation.

In section 9 we have seen how normalized nonnegative models and nonnegative matrix factorization (NMF) are related to each other. NMF plays an important role in recommendation in general; see [Ma et al., 2011]. Of course, probabilistic models can be regarded as a conveniently regularized NMF but that perspective disregards the operational interpretation of the columns of the nonnegative matrices  $A, B$  that constitute the NMF  $A^T B$ . This interpretation is important and the choice of regularization of an NMF affects the predictive performance of the studied recommender system.

## 11 Conclusion

We evaluated normalized nonnegative models in the context of item recommendation and we extended our understanding of these models; both from the practical and theoretical perspective. We deepened the practical understanding of normalized nonnegative models by studying their performance in top- $N$  recommendation and by introducing user-user and item-item similarity measures which can be interpreted operationally in terms of the distinguishability of users and the distinguishability of items. On the theoretical side we extended our understanding of normalized nonnegative models *by* showing how the regularization scheme defining normalized nonnegative models can enforce unique user and item representations, *by* proving that the inference of optimal normalized nonnegative models is *NP*-hard and *by* explaining how normalized nonnegative models are related to nonnegative matrix factorizations.

## 12 Acknowledgments

I thank Robin Kothari, Patrick Pletscher and Sharon Wulff for interesting and fruitful discussions. I thank the authors of [Cremonesi et al., 2010, Barbieri and Manco, 2011] for providing the source files for the figures in [Cremonesi et al., 2010, Barbieri and Manco, 2011]. I acknowledge funding by the ARO grant Contract Number W911NF-12-0486. This work is preprint MIT-CTP/4738.

## A Application of item-item similarity: characterization of stereo-types and hierarchical orderings

In [Stark, 2015] we explained how normalized nonnegative models can be used to succinctly describe user and items in terms of interpretable tags of items. This requires that the available data specifies tags for each item. What if no tags are available? For these circumstances we suggest using  $k$ -medoids to classify the items with respect to the item-item similarity measure (11). Here,  $k = G$ , i.e.,  $k$  equals the number of (effective) tags we want to use. Running  $k$ -medoids thus assigns tags  $g \in [G]$  to items. Using these tags we can proceed as in sections 5.1 and 6 from [Stark, 2015].

The tags we compute by  $k$ -medoids do not, however, come along with intuitive names like *comedy*, *drama* or *romance*. Hence, to intuitively understand the effective tags we propose selecting popular representative items for each of the tags (e.g., movies the user knows already). The computed tags can then be described to users in terms of the representative items. For example, in movie recommendation where tags specify genres, effective  $\text{genre}_3 \sim \{\text{movie}_{97}, \text{movie}_{17}, \text{movie}_{97}\}$ .

## B Alternative interpretation of $\delta(\vec{p}_1, \vec{p}_2)$

In collaborative filtering we oftentimes recommend an item  $i$  to user  $u_2$  if  $u_1$  rated and liked item  $i$ , and if  $u_1$  and  $u_2$  are similar users. Hence, the probability that users  $u_1$  and  $u_2$  provide different ratings for  $i$  is crucial for us. This probability can be lower bounded as follows.

$$\begin{aligned} \mathbb{P}[R_{u_1 i} \neq R_{u_2 i}] &= 1 - \mathbb{P}[R_{u_1 i} = R_{u_2 i}] = 1 - \sum_z \mathbb{P}[R_{u_1 i} = z, R_{u_2 i} = z] \\ &= 1 - \sum_z \mathbb{P}_1[\hat{E}_i = z] \mathbb{P}_2[\hat{E}_i = z] \geq 1 - \sum_z \min\{\mathbb{P}_1[\hat{E}_i = z], \mathbb{P}_2[\hat{E}_i = z]\} \\ &= \delta(\vec{p}_1, \vec{p}_2). \end{aligned} \tag{14}$$

## C Error measures

When evaluating the performance of recommender systems, the choice for quantifying the prediction error crucially affects the evaluation; the observation that a system  $A$  performs better than a system  $B$  oftentimes changes if we change the particular way we quantify the prediction error.

*Root-mean-squared-error* (RMSE) and *mean-average-error* (MAE) are popular choices for measuring the prediction error. If we denote by  $\hat{R}_{ui}$  the rating predicted by our recommendation system, then  $\text{RMSE} = (\sum_{(ui) \in T} (R_{ui} - \hat{R}_{ui})^2) / |T|$  and  $\text{MAE} = (\sum_{(ui) \in T} |R_{ui} - \hat{R}_{ui}|) / |T|$ . It follows that RMSE and MAE can be regarded as  $l_2$ -distance and  $l_1$ -distance between prediction and ground truth. Being an instance of an  $l_2$ -type distance, RMSE is sensitive to outliers in  $(\hat{R}_{ui})_{ui \in T}$  whereas MAE is not. In practice, however, it is oftentimes not of immediate interest to predict actual ratings. Instead we are interested in presenting to the user a short list of items that are of interest to that user. This short list of recommendations is the *top- $N$  recommendation*. *Precision* and *recall* are error measures designed to measure the usefulness of these top- $N$  recommendations that are computed by recommender systems. Following [Cremonesi et al., 2010], we define

recall and precision through the following procedure. Fix  $N$ . Then, for each triple  $(u, i, R_{ui}) \in T$  satisfying  $R_{ui} = 5$ ,

1. sample 1000 items not rated by user  $u$ .
2. Using the recommender system under evaluation, compute predictions for the ratings of user  $u$  for  $i$  and for the 1000 random items from step 1.
3. Sort the 1001 items under consideration descendingly according to their predicted ratings.
4. Denote by  $p$  the position of item  $i$  in that sorted list.
5. Define a top- $N$  recommendation by selecting the first  $N$  items from the sorted list.
6. If  $p \leq N$  then we have a *hit*. Else we have a *miss*. Thus, for each entry  $(u, i, R_{ui}) \in T$  satisfying  $R_{ui} = 5$  we either get a hit or a miss.

Then, *recall at  $N$*  is the average number of hits for  $T$ , i.e.,

$$\text{recall}(N) := \frac{\text{number of hits}}{|T|}.$$

A closely related quantity is *precision at  $N$* . Precision specifies how much recall we have per item in the top- $N$  recommendation list, i.e.,

$$\text{precision}(N) := \frac{\text{recall}(N)}{N}.$$

## D Items from the long-tail

Data available to train recommender systems usually follows a *long-tail* distribution. I.e., a vast majority of the ratings available for training are ratings of a tiny fraction of all the items. For instance in the MovieLens 1M dataset, 5.5% (i.e., 213 movies) of the most popular items amount for 33% of all the ratings. As a user we might be a little disappointed by recommendations of very popular items as we may already be aware of those items. On the other hand, as a provider of the items, we want to push diversity in our product line. This motivated the testing methodology employed in [Cremonesi et al., 2010] where the most popular items (6%) are removed from the test set  $T$ .

## E Proof of Theorem 1

The *decision version* of *MinDim* is  $NNM_D$ .

$NNM_D$ . *Problem instance*:  $(\mathbb{P}_u[\hat{E}_i = z])_{uiz \in \Omega}$  for some  $\Omega \subseteq [U] \times [I] \times [Z]$  marking the probabilities that are known a priori. *Acceptance condition*: we output *yes* if and only if there exists a  $D$ -dimensional normalized nonnegative model (NNM)  $\vec{p}_u, \vec{E}_{iz}$  such that  $\vec{p}_u^T \vec{E}_{iz} = \mathbb{P}_u[\hat{E}_i = z]$  for all  $(u, i, z) \in \Omega$ .

Here we prove the following Theorem 3; it implies that *MinDim* is *NP*-hard because  $NNM_D$  has to be accepted if and only if the minimizer of *MinDim* is  $\leq D$ .

**Theorem 3.** *The decision problem  $NNM_D$  is NP-hard.*

We prove Theorem 3 in terms of a reduction from *EXACT NMF<sub>k</sub>* (see [Vavasis, 2009]) to  $NNM_D$ ;

*EXACT NMF<sub>k</sub>* (see [Vavasis, 2009]). *Problem instance*: a nonnegative matrix  $M \in \mathbb{R}_+^{m \times n}$  with  $\text{rank}(M) = k$ . *Acceptance condition*: we output *yes* if and only if there exist nonnegative matrices  $W \in \mathbb{R}_+^{k \times m}$ ,  $H \in \mathbb{R}_+^{k \times n}$  with  $k := \text{rank}(M)$  such that  $M = W^T H$ .

A reduction from *EXACT NMF<sub>k</sub>* to  $NNM_D$  suffices to prove the theorem because *EXACT NMF<sub>k</sub>* is *NP*-hard;

**Theorem 4** (see [Vavasis, 2009]). *The decision problem EXACT NMF<sub>k</sub> is NP-hard.*

To prove theorem 3 we show that there exists a polynomial time algorithm  $\mathcal{A}$  with the two properties

$$\mathcal{A} : \{\text{instances } M \text{ for EXACT NMF}_k\} \rightarrow \{\text{instances for NNM}_k\}$$

and

$$\mathcal{A}(M) \text{ yes for NNM}_k \Leftrightarrow \text{yes for EXACT NMF}_k. \quad (15)$$

The algorithm  $\mathcal{A}$  we employ here does the following (recall that  $M \in \mathbb{R}_+^{m \times n}$ ).

- Compute  $M' \in \mathbb{R}_+^{m \times n}$  by normalizing each row  $M_{i,:}$  of  $M$ , i.e.,

$$M'_{i,:} = M_{i,:} / \left( \sum_j M_{ij} \right).$$

- Set  $U = m$ ,  $I = 1$ ,  $Z = n$  and

$$\mathbb{P}_u[\hat{E}_1 = z] = M'_{uz}.$$

- Output  $(\mathbb{P}_u[\hat{E}_1 = z])_{uiz \in \Omega}$  with  $\Omega = [m] \times [1] \times [n]$ .

We recognize that  $\mathcal{A}$  indeed maps instances for EXACT NMF<sub>k</sub> to instances for NNM<sub>k</sub>. It is left to show (15).

“ $\Rightarrow$ ”: by assumption there exist  $k$ -dimensional distributions  $\vec{p}_u$  and rating vectors  $\vec{E}_{1z}$  such that  $\vec{p}_u^T \vec{E}_{1z} = M'_{uz}$ . Hence, the matrices  $(\vec{p}_u, \dots, \vec{p}_U)$  and  $(\vec{E}_{11}, \dots, \vec{E}_{1Z})$  realize the wanted  $k$ -dimensional NMF.

“ $\Leftarrow$ ”: by assumption there exist  $W \in \mathbb{R}_+^{k \times m}$ ,  $H \in \mathbb{R}_+^{k \times n}$  such that  $M = W^T H$ . Therefore, setting

$$W' := W \text{diag}\left(1/\left(\sum_j M_{1j}\right), \dots, 1/\left(\sum_j M_{mj}\right)\right)$$

and  $H' := H$ , we get  $M' = W'^T H'$ . We denote by  $W'_{:,u}$  and  $H'_{:,z}$  columns of  $W'$  and  $H'$ , respectively. Set  $\vec{\eta} = \sum_z H'_{:,z}$  and  $T = \text{diag}(\vec{\eta})$ . Then,  $\vec{E}_{1z} := T^{-1} H'_{:,z}$  returns valid rating vectors in  $\mathcal{E}'$ . Moreover, the ansatz  $\vec{p}_u := T W'_{:,u}$  yields valid probability distributions because

$$\|\vec{p}_u\|_1 = \vec{p}_u^T (1, \dots, 1)^T = \vec{p}_u^T \left( \sum_z \vec{E}_{1z} \right) = (T W'_{:,u})^T \left( \sum_z T^{-1} H'_{:,z} \right) = \sum_z M'_{uz} = 1.$$

This proves the claim because  $\vec{p}_u^T \vec{E}_{1z} = W'_{:,u}^T H'_{:,z} = M'_{uz} = \mathbb{P}_u[\hat{E}_1 = z]$ .

## F Proof of Lemma 2

Set

$$M := (\mathbb{P}_u[\hat{E}_i = z])_{u \in [U]; i \in [I], z \in [Z]} \in \mathbb{R}^{U \times IZ}.$$

so that  $M = A^T B$ . By  $D = \text{rank}(A)$ , there exist  $u_1, \dots, u_D$  such that the columns  $\{\vec{a}_{u_k}\}_{k=1}^D$  form a basis for  $\mathbb{R}^D$ . By normalization of probability distributions,

$$1 = \sum_z \mathbb{P}_{u_k}[\hat{E}_i = z] = \vec{a}_{u_k}^T \left( \sum_z \vec{b}_{iz} \right)$$

for all  $k \in [D]$ . It follows that for all  $i, i' \in [I]$

$$\sum_z \vec{b}_{iz} = \sum_z \vec{b}_{i'z} =: \vec{\eta} \in \mathbb{R}_+^D$$

because  $\{\vec{a}_{u_k}\}_{k=1}^D$  is a basis. Assume that  $\eta_j > 0$  for all  $j$  (this will be proven afterwards). Then,  $T := \text{diag}(\vec{\eta})$  is invertible. Hence,  $B' := T^{-1}B$  is well defined and the columns  $\vec{b}'_{iz}$  of  $B'$  satisfy

$$\sum_z \vec{b}'_{iz} = T^{-1} \left( \sum_z \vec{b}_{iz} \right) = (1, \dots, 1)^T$$

because  $T^{-1}\vec{\eta} = (1, \dots, 1)^T$ . Consequently, the vectors  $\vec{b}'_{iz}$  satisfy the normalization condition (2). Moreover, the nonnegative columns  $\vec{a}'_u$  are valid probability distributions because

$$\|\vec{a}'_u\|_1 = \vec{a}'_u{}^T (1, \dots, 1)^T = \vec{a}'_u{}^T \left( \sum_z \vec{b}'_{iz} \right) = (T\vec{a}_u)^T \left( \sum_z T^{-1}\vec{b}_{iz} \right) = \sum_z \mathbb{P}_u[\hat{E}_i = z] = 1.$$

Therefore, the vectors  $\vec{a}'_u$  and  $\vec{b}'_{iz}$  constitute a valid NNM. This almost concludes the proof of the Lemma because

$$\vec{a}'_u{}^T \vec{b}'_{iz} = \vec{a}_u{}^T \vec{b}_{iz} = \mathbb{P}_u[\hat{E}_i = z]$$

for all  $u, i, z$ . It only remains to verify that  $\eta_j \neq 0$  for all  $j$ . We provide an argument that is similar to an argument from [Lee et al., 2014] which was used to prove a relation between general positive semidefinite factorizations and quantum models. Assume that there exists  $j$  such that  $\eta_j = 0$ . Thus,

$$0 = \eta_j = \left( \frac{1}{I} \sum_{iz} \vec{b}_{iz} \right)_j. \quad (16)$$

Now assume there exists  $i, z$  with  $(\vec{b}_{iz})_j = \varepsilon > 0$ . Then,  $\sum_{iz} (\vec{b}_{iz})_j \geq \varepsilon > 0$  because  $(\vec{b}_{iz})_j \geq 0$  for all  $i, z$ . This contradicts (16) and therefore,  $\eta_j = 0$  implies  $(\vec{b}_{iz})_j = 0$  for all  $i, z$ . That, however, violates the condition  $D = \text{rank}(B)$  from the Lemma. We conclude that there cannot exist  $j$  with  $\eta_j = 0$ .

## References

- [Barbieri and Manco, 2011] Barbieri, N. and Manco, G. (2011). An analysis of probabilistic methods for top-N recommendation in collaborative filtering. *Machine Learning and Knowledge Discovery in Databases*, pages 172–187.
- [Blei et al., 2004] Blei, D. M., Griffiths, T. L., Jordan, M. I., and Tenenbaum, J. B. (2004). Hierarchical topic models and the nested chinese restaurant process. In *Advances in Neural Information Processing Systems*.
- [Blei et al., 2003] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022.
- [Cover and Thomas, 2012] Cover, T. M. and Thomas, J. A. (2012). *Elements of information theory*. John Wiley & Sons.
- [Cremonesi et al., 2010] Cremonesi, P., Koren, Y., and Turrin, R. (2010). Performance of recommender algorithms on top-N recommendation tasks. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 39–46. ACM.
- [Deshpande and Karypis, 2004] Deshpande, M. and Karypis, G. (2004). Item-based top-N recommendation algorithms. *ACM Transactions on Information Systems (TOIS)*, 22(1):143–177.
- [Herlocker et al., 2002] Herlocker, J., Konstan, J. A., and Riedl, J. (2002). An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Information retrieval*, 5(4):287–310.

- [Herlocker et al., 2004] Herlocker, J. L., Konstan, J. A., Terveen, L. G., and Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1):5–53.
- [Hofmann, 1999] Hofmann, T. (1999). Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 289–296. Morgan Kaufmann Publishers Inc.
- [Hofmann and Puzicha, 1999] Hofmann, T. and Puzicha, J. (1999). Latent class models for collaborative filtering. In *IJCAI*, volume 99, pages 688–693.
- [Koren et al., 2009] Koren, Y., Bell, R., and Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, (8):30–37.
- [Lee and Seung, 1999] Lee, D. D. and Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791.
- [Lee et al., 2014] Lee, T., Wei, Z., and de Wolf, R. (2014). Some upper and lower bounds on psd-rank. *arXiv preprint arXiv:1407.4308*.
- [Ma et al., 2011] Ma, H., Zhou, D., Liu, C., Lyu, M. R., and King, I. (2011). Recommender systems with social regularization. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 287–296. ACM.
- [McLaughlin and Herlocker, 2004] McLaughlin, M. R. and Herlocker, J. L. (2004). A collaborative filtering algorithm and evaluation metric that accurately model the user experience. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 329–336. ACM.
- [Miller et al., 2003] Miller, B. N., Albert, I., Lam, S. K., Konstan, J. A., and Riedl, J. (2003). MovielenS unplugged: experiences with an occasionally connected recommender system. In *Proceedings of the 8th international conference on Intelligent user interfaces*, pages 263–266. ACM.
- [Mnih and Salakhutdinov, 2007] Mnih, A. and Salakhutdinov, R. (2007). Probabilistic matrix factorization. In *Advances in neural information processing systems*, pages 1257–1264.
- [O Connor and Herlocker, 1999] O Connor, M. and Herlocker, J. (1999). Clustering items for collaborative filtering. In *Proceedings of the ACM SIGIR workshop on recommender systems*, volume 128. Citeseer.
- [Resnick et al., 1994] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. (1994). Grouplens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pages 175–186. ACM.
- [Salakhutdinov and Mnih, 2008] Salakhutdinov, R. and Mnih, A. (2008). Bayesian probabilistic matrix factorization using markov chain monte carlo. In *Proceedings of the 25th international conference on Machine learning*, pages 880–887. ACM.
- [Sarwar et al., 2001] Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295. ACM.
- [Sarwar et al., 2002] Sarwar, B. M., Karypis, G., Konstan, J., and Riedl, J. (2002). Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering. In *Proceedings of the fifth international conference on computer and information technology*, volume 1. Citeseer.
- [Stark, 2015] Stark, C. (2015). Expressive recommender systems through normalized nonnegative models. *accepted to AAAI-16 conference; arXiv preprint arXiv:1511.04775*.
- [Vavasis, 2009] Vavasis, S. A. (2009). On the complexity of nonnegative matrix factorization. *SIAM Journal on Optimization*, 20(3):1364–1377.